ADAPTING ASR FOR UNDER-RESOURCED LANGUAGES USING MISMATCHED TRANSCRIPTIONS

Chunxi Liu†, Preethi Jyothi*,‡, Hao Tang†, Vimal Manohar†, Rose Sloan¶, Tyler Kekona∥, Mark Hasegawa-Johnson§, Sanjeev Khudanpur†

† Center for Language and Speech Processing, Johns Hopkins University, MD USA
‡ Beckman Institute, University of Illinois at Urbana-Champaign, IL USA
¶ Yale University, CT USA
∥ Toyota Technological Institute at Chicago, IL USA
∥∥ University of Washington, WA USA

ABSTRACT

Mismatched transcriptions of speech in a target language refers to transcriptions provided by people unfamiliar with the language, using English letter sequences. In this work, we demonstrate the value of such transcriptions in building an ASR system for the target language. For different languages, we use less than an hour of mismatched transcriptions to successfully adapt baseline multilingual models built with no access to native transcriptions in the target language. The adapted models provide up to 25% relative improvement in phone error rates on an unseen evaluation set.

Index Terms— mismatched transcriptions, ASR adaptation, ASR for under-resourced languages

1. INTRODUCTION

Speech and language technologies are unavailable to a large majority of the world’s languages. Most languages are under-resourced in terms of the technological resources needed to build speech recognition systems (see [1] for a detailed survey on speech processing for under-resourced languages). A first step towards building an ASR system for a new language typically involves collecting sufficient amounts of transcribed speech data. Crowdsourcing has been explored as an innovative way in which transcriptions for speech data are solicited from large numbers of crowd workers who are native speakers of the language [2, 3]. This technique, however, would be constrained to languages for which it is possible to find native speakers online. To circumvent the need for native transcribers in a language, mismatched crowdsourcing has been introduced as a technique [4, 5] that makes use of transcriptions in the form of English syllables from crowd workers unfamiliar with the target language (henceforth referred to as “mismatched transcriptions”).

Mismatched transcriptions have been demonstrated to produce reasonably accurate transcriptions in both isolated word and continuous speech tasks. Using mismatched transcriptions, isolated Hindi words from a medium vocabulary task were recovered with a 1-best accuracy of over 85% [4] and transcriptions for short continuous speech segments in Hindi were labeled with an accuracy of over 55% on a large vocabulary task. A logical question that follows is: what is the impact of mismatched transcriptions on ASR performance? This is the main question that is tackled in this work.

There has been a lot of prior work on building speech technologies for under-resourced languages using multilingual models from high-resource languages and applying unsupervised adaptation techniques (e.g. [6] for Polish, [7] for Vietnamese and [8] for Czech). Our work is the first attempt to explore the use of mismatched transcriptions as adaptation data with multilingual models. We also demonstrate significant performance improvements from using mismatched transcriptions beyond that obtained from adaptation with untranscribed speech data in the target language.

2. PROBLEM SETUP

Our goal is to train a phone recognition system for a given target language in which no native transcriptions are available. We assume that we have access to unspoken texts and to untranscribed audio in the target language, but not to transcribed audio. Baseline multilingual systems are trained using native transcriptions from several different languages (not including the target language). Section 4 details multilingual GMM-HMM and DNN-based ASR systems with language-specific grammar models and Section 5 describes a semi-supervised baseline that uses unlabeled data from the target language. Next, we adapt the parameters of the acoustic model of the above system using only probabilistic phone transcriptions in the target language derived from mismatched transcriptions. The construction of probabilistic phone transcriptions is described in Section 3 and the acoustic model adaptation is detailed in Section 6.

2.1. Task Details

Our speech data were extracted from publicly available Special Broadcasting Service Australia (SBS) radio podcasts [9] hosted in 68 different languages. We restricted our experiments to seven of these languages for which we could find a native transcriber willing to provide orthographic transcriptions for roughly 1 hour of speech: Arabic (AR), Cantonese (CA), Dutch (DT), Hungarian (HG), Mandarin (MD), Swahili (SW) and Urdu (UR).1

The SBS radio podcasts are not entirely homogeneous in the target language and contain utterances interspersed with segments of music and English. A simple GMM-based language identification system was developed as a first pass over the podcasts in order to isolate regions that correspond mostly to the target language. These long segments were then split into smaller $\approx$ 5-second segments. This was to enable easy labeling by the native transcribers, and more

1CA transcriptions were provided by Nancy Chen at $I^2$R in Singapore, as part of a collaborative research project. The native transcribers for the other six languages were paid student volunteers at the University of Illinois at Urbana-Champaign.
Table 1. 1-best probabilistic phone transcription error rates on the development and evaluation sets.

<table>
<thead>
<tr>
<th>Language Code</th>
<th>Dev set (1-best)</th>
<th>Eval set (1-best)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR</td>
<td>65.8</td>
<td>66.2</td>
</tr>
<tr>
<td>CA</td>
<td>66.4</td>
<td>67.8</td>
</tr>
<tr>
<td>DT</td>
<td>68.9</td>
<td>70.9</td>
</tr>
<tr>
<td>HG</td>
<td>63.7</td>
<td>63.5</td>
</tr>
<tr>
<td>MD</td>
<td>70.9</td>
<td>69.6</td>
</tr>
<tr>
<td>SW</td>
<td>47.6</td>
<td>50.3</td>
</tr>
<tr>
<td>UR</td>
<td>67.2</td>
<td>70.5</td>
</tr>
</tbody>
</table>

Fig. 1. Phone error rates plotted against entropy rate estimates of phone sequences in three different languages.

- $\Pr(\lambda | \pi)$ is trained using the Carmel toolkit [11] as a probabilistic finite state transducer (FST) mapping phones to letters. We also allow this FST to delete phones and insert letters. The training uses representative transcripts $\lambda$ and their corresponding phone transcripts $\pi$ (derived from orthographic transcripts, as described in Section 4), for speech in six languages other than the target language. We assume that such a model approximates $\Pr(\lambda | \pi)$ for the target language. Note that, while this assumption is not entirely accurate, it is necessitated by the requirement that no native transcriptions in the target language can be used in building any part of our system.

A crude measure of the quality of the PTs is given by the phone error rate between $\pi^* = \arg \max_\pi \Pr(\pi | T)$ and the reference phone sequences. Table 1 lists these 1-best error rates on the SBS development and evaluation sets, for all seven languages. However, the 1-best error rates do not accurately reflect the extent of information in the PTs that can be leveraged during ASR adaptation. A fuller picture is obtained by considering a collection of sequences $\Pi$ that are almost as probable as $\pi^*$ according to our model. Figure 1 shows the trend of phone error rates (for three languages) obtained by using collections $\Pi$ of increasing size, plotted against an entropy estimate of $\Pi$. This estimate measures the average entropy of phones in the sequences in $\Pi$; e.g., 1 bit of entropy allows two equally probable choices for each phone in $\pi$. We note that the phone error rates significantly drop across all languages, staying within 1 bit of entropy per phone, illustrating the extent of information captured by the PTs.

4. MULTILINGUAL BASELINES

The goal of building a multilingual system is two-fold. One is to setup a baseline for generalizing to an unseen language without any labeled audio corpus. The other is to have the baseline serve as a starting point for adaptation.

The dataset consists of 40 minutes of labeled audio for training,
of the self-training approach described in [19]. The multilingual DNN trained in Section 4 is used as a seed model to decode the unlabeled audio. The phone language model used during decoding was trained on target-language text. Lattice posteriors are used as confidence measures and only frames with posteriors $\geq 0.7$ on the best path of the lattice are selected. We empirically determined that it was better to use the posteriors as soft-targets during frame-based cross-entropy training. This is different from the approach in [19], which used the best path alignment as the target. Additionally, we scaled the amount of transcribed data by 2 to create a good balance between transcribed and unlabeled data as suggested in [19].

The results using these semi-supervised DNN-HMM systems are shown in Table 3. Although, semi-supervised training improves PER performance compared to the multilingual DNN-HMMs, it still falls short of the improvements obtained by adaptation with probabilistic transcriptions (described in Section 6). This is in spite of the unlabeled audio data being several times larger than the probabilistic transcription data.

6. ASR ADAPTATION USING PTS

As can be seen from Table 2, the multilingual baseline systems appear not to generalize well to an unseen target language. This section will detail how we improve the generalization capability of these multilingual systems to an unseen target language using mismatched probabilistic transcriptions (described in Section 3).

Our ASR framework is based on weighted finite-state transducers (WFSTs) as outlined in [20]. In this framework, the acoustic model is specified by a probabilistic mapping from acoustic signals to a sequence of discrete symbols, and a WFST $H$ mapping these symbol sequences to triphone sequences. The other WFSTs in the framework are $C$ which maps down triphone sequences to monophone sequences, a pronunciation model $L$ and a language model $G$. Since our tasks involve phone recognition, $L$ is essentially an identity mapping and $G$ is a phone N-gram model.

To describe the adaptation process, it will be helpful to compare the following two cases.

- In training the parameters of the baseline acoustic model, for each training utterance, we work with the cascade $H \circ C \circ L \circ T$, where $T$ is a linear chain FST representing the training transcript. The multilingual baselines described in Section 4 are trained in this manner using training data from languages other than the target language.
- During adaptation, for each training utterance (in the target language), we work with the cascade $H \circ C \circ L \circ PT$, where $PT$ is a WFST representing the probabilistic transcript, obtained as in Section 3.

As noted in Figure 1, a PT contains significant amount of information beyond any single transcript extracted from the PT. Motivated by this, the statistics for the MAP estimation are accumulated from a lattice derived from the cascade $H \circ C \circ L \circ PT$.

6.1. MAP estimation of the acoustic model

The Bayesian framework for maximum a posteriori (MAP) estimation has been widely applied to GMM and HMM parameter estimation problems such as parameter smoothing and speaker adaptation [21].

Formally, for an unseen target language, we denote its acoustic observations $x = (x_1, \ldots, x_T)$, and its acoustic model parameter
from the ones used in typical Kaldi recipes, to determinize these items. We use two additional disambiguation symbols, apart of phones and insertions of letters (to a maximum of 3 in our experiments). An additional WFST restricting the number of consecutive deletions is applied to the cascade for purposes of computational efficiency, the cascade for \( \text{proj} \). The cascade of WFSTs modeling the distributions shown in Equation 1 is an acceptor defined as \( H \).

The baseline and the adapted models were implemented using HG & SW, with a re-alignment stage in iteration 10. For this work, we demonstrate the utility of mismatched transcriptions in significantly improving ASR systems for different target languages. A relative reduction of phone error rate of up to 25% (for Swahili) is observed on adapting baseline ASR systems using mismatched transcriptions from the target language. We observe a consistent drop in error rates moving from MUL-L to SS. PT-ADAPT corresponds to PERs from the multilingual GMM-HMM systems adapted to mismatched transcriptions from the target language. We observe substantial PER improvements with PT-ADAPT over MUL-L across all four languages. We also find that PT adaptation consistently outperforms the SS systems for all four languages. (The relative reductions in PER compared to both baselines are listed in the last two columns.) This suggests that adaptation with PTs is providing more information than that obtained by model self-training alone. It is also interesting that we obtain significantly larger PER improvements with PTs for Swahili compared to the other three languages. We conjecture this may be partly because Swahili has a relatively simple phonology (compared to the other three languages) and most consonants have almost the same pronunciation as in English [23].

### 6.2. Implementation details

The baseline and the adapted models were implemented using Kaldi [22]. In order to efficiently carry out the required operations on the cascade \( H \circ C \circ L \circ PT \), we carefully design \( PT \). \( PT \) is an acceptor defined as \( \text{proj}_{\text{input}}(PT) \) where \( PT \) is a WFST mapping phone sequences to English letter sequences obtained as a cascade of WFSTs modeling the distributions shown in Equation 1 and \( \text{proj}_{\text{input}} \) refers to projecting onto the input labels. For the purposes of computational efficiency, the cascade for \( PT \) includes an additional WFST restricting the number of consecutive deletions of phones and insertions of letters (to a maximum of 3 in our experiments). We use two additional disambiguation symbols [20], apart from the ones used in typical Kaldi recipes, to determine these insertions and deletions in \( PT \). MAP adaptation for acoustic model was carried out for a number of iterations (12 for CA & MD, 14 for HG & SW, with a re-alignment stage in iteration 10).

### 6.3. Experimental results

Table 3 presents phone error rates (PERs) on the evaluation (and development) sets for four different languages. MULT-L corresponds to the multilingual GMM-HMM baseline error rates reproduced from Table 2 and SS refers to the DNN-HMM multilingual baselines adapted with untranscribed audio in the target language. We observe a consistent drop in error rates moving from MULT-L to SS. PT-ADAPT corresponds to PERs from the multilingual GMM-HMM systems adapted to mismatched transcriptions from the target language. We observe substantial PER improvements with PT-ADAPT over MULT-L across all four languages. We also find that PT adaptation consistently outperforms the SS systems for all four languages. (The relative reductions in PER compared to both baselines are listed in the last two columns.) This suggests that adaptation with PTs is providing more information than that obtained by model self-training alone. It is also interesting that we obtain significantly larger PER improvements with PTs for Swahili compared to the other three languages. We conjecture this may be partly because Swahili has a relatively simple phonology (compared to the other three languages) and most consonants have almost the same pronunciation as in English [23].

### 7. CONCLUSIONS

In this work, we demonstrate the utility of mismatched transcriptions in significantly improving ASR systems for different target languages. A relative reduction of phone error rate of up to 25% (for Swahili) is observed on adapting baseline ASR systems using mismatched transcriptions. Similar impact is shown for languages from different language families and containing sounds with distinctive phonological properties.

### 8. ACKNOWLEDGMENTS

The work reported here was started at JSALT 2015 in UW, Seattle, and was supported by JHU via grants from NSF (IIS), DARPA (LORELEI), Google, Microsoft, Amazon, Mitsubishi Electric, and MERL. The authors also thank Amit Das at the University of Illinois for his help with preprocessing the SBS audio podcasts.

---

### Table 2: PERs of unadapted multilingual systems and monolingual systems on the evaluation and development sets (latter within parentheses).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CA</td>
<td>79.64 (79.83)</td>
<td>77.13 (77.85)</td>
<td>83.28 (82.12)</td>
<td>82.99 (81.86)</td>
<td>32.77 (34.61)</td>
<td>27.67 (28.88)</td>
</tr>
<tr>
<td>HG</td>
<td>78.62 (77.58)</td>
<td>75.98 (76.44)</td>
<td>81.86 (80.47)</td>
<td>82.30 (81.18)</td>
<td>39.58 (39.77)</td>
<td>35.87 (36.58)</td>
</tr>
<tr>
<td>MD</td>
<td>68.40 (68.35)</td>
<td>68.62 (66.90)</td>
<td>71.30 (68.66)</td>
<td>63.04 (64.73)</td>
<td>32.21 (26.92)</td>
<td>27.80 (23.96)</td>
</tr>
<tr>
<td>SW</td>
<td>66.54 (65.28)</td>
<td>66.08 (66.58)</td>
<td>65.77 (64.80)</td>
<td>64.75 (65.04)</td>
<td>35.33 (46.51)</td>
<td>34.98 (41.47)</td>
</tr>
</tbody>
</table>

### Table 3: PERs on the evaluation and development sets (latter within parentheses) before and after adaptation with PTs.

<table>
<thead>
<tr>
<th>Language Code</th>
<th>(MULT-L)</th>
<th>(SS)</th>
<th>MULT-L + PT adaptation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA</td>
<td>68.40 (68.35)</td>
<td>63.79 (62.46)</td>
<td>57.20 (56.57)</td>
</tr>
<tr>
<td>HG</td>
<td>68.62 (66.90)</td>
<td>63.53 (63.50)</td>
<td>56.98 (57.26)</td>
</tr>
<tr>
<td>MD</td>
<td>71.30 (68.66)</td>
<td>64.90 (64.00)</td>
<td>58.21 (57.85)</td>
</tr>
<tr>
<td>SW</td>
<td>63.04 (64.73)</td>
<td>58.76 (59.81)</td>
<td>44.31 (48.88)</td>
</tr>
</tbody>
</table>

---

set as \( \lambda_{\text{MAP}} \), then the MAP parameters are defined as:

\[
\lambda_{\text{MAP}} = \arg\max_{\lambda} \Pr(\lambda|x) = \arg\max_{\lambda} \Pr(x|\lambda) \Pr(\lambda) \tag{2}
\]

where we use multilingual baseline GMM-HMM parameters to assign the conjugate prior hyperparameters in \( p(\lambda) \), and take the modes of the prior distributions as the initial model parameter estimates. Using suitable models for these distributions, [21] derive update rules in an EM algorithm for computing \( \lambda_{\text{MAP}} \). For example, the mean \( \mu_{ik} \) of the GMM mixture component \( k \) associated with HMM state \( i \) is updated as:

\[
\hat{\mu}_{ik} = \frac{\tau_{ik} \pi_{ik} + \alpha_{ik} \hat{\mu}_{ik}}{\tau_{ik} + \alpha_{ik}} \tag{3}
\]

\[
\alpha_{ik} = \sum_{t=1}^{T} c_{ikt}
\]

\[
\tau_{ik} = \frac{1}{\sum_{t=1}^{T} c_{ikt}}
\]

\[
\hat{\mu}_{ik} = \frac{\sum_{t=1}^{T} c_{ikt} x_{it}}{\sum_{t=1}^{T} c_{ikt}}
\]

where \( \tau_{ik} \) is a hyperparameter in the prior density for the mixture component \( k \) of state \( i \) and \( c_{ikt} \) denotes the probability of the HMM being in state \( i \) with mixture component \( k \) given observation \( x_{it} \) (estimated using statistics accumulated from the cascade \( H \circ C \circ L \circ PT \)). In our setting, the initial value of \( \mu_{ik} \) is obtained from the multilingual baseline model, and \( \hat{\mu}_{ik} \) eventually converges to a model for the target language data.
9. REFERENCES


